Scientific Research Publishing

Journal of Applied Mathematics and Physics, 2022, 10, 1619-1630
https://www.scirp.org/journal/jamp
ISSN Online: 2327-4379
ISSN Print: 2327-4352

Lagrangian Relaxation Method for Multiobjective Optimization Methods: Solution Approaches

H. S. Faruque Alam

Department of Mathematics, University of Chittagong, Chittagong, Bangladesh
Email: hsfa@cu.ac.bd

How to cite this paper: Alam, H.S.F. (2022) Lagrangian Relaxation Method for Multiobjective Optimization Methods: Solution Approaches. *Journal of Applied Mathematics and Physics*, **10**, 1619-1630.
https://doi.org/10.4236/jamp.2022.105112

Received: April 19, 2022
Accepted: May 22, 2022
Published: May 25, 2022

Abstract

This paper introduces the Lagrangian relaxation method to solve multiobjective optimization problems. It is often required to use the appropriate technique to determine the Lagrangian multipliers in the relaxation method that leads to finding the optimal solution to the problem. Our analysis aims to find a suitable technique to generate Lagrangian multipliers, and later these multipliers are used in the relaxation method to solve Multiobjective optimization problems. We propose a search-based technique to generate Lagrange multipliers. In our paper, we choose a suitable and well-known scalarization method that transforms the original multiobjective into a scalar objective optimization problem. Later, we solve this scalar objective problem using Lagrangian relaxation techniques. We use Brute force techniques to sort optimum solutions. Finally, we analyze the results, and efficient methods are recommended.

Keywords

Multiobjective Optimization Problem, Lagrangian Relaxation, Lagrange Multipliers, Scalarization Method

1. Introduction

Many real-life problems turn into nonsmooth and large-scale when formulated mathematically. There are not enough approaches to solve these nonsmooth and large-scale problems. In such a case, the Lagrangian relaxation is used in dealing with the problems when they are nonsmooth and difficult to solve. The method can be considered as the extended nonlinear programming Lagrange multiplier method. Theoretical concepts of the Lagrangian relaxation method introduced in 1970's considered a tool that is the backbone of a number of large-scale applications. Several surveys of the Lagrangian relaxation method were carried out by Fisher [1], Geoffrion [2], Shapiro [3]. Later, Fisher [1] and Held *et al.* [4] provided methodologies to use the Lagrangian relaxation approach in practice.

Many difficult integer problems can be altered into relatively simple problems when Lagrangian relaxation is used. To exploit this transformation, we create a Lagrangian problem in which the complicating constraints are replaced with a penalty term in the objective function involving the amount of violation of the constraints and their dual variables. The Lagrangian problem is relatively easy to solve and provides a lower bound (for minimization problem) on the optimal value of the original problem. It can be used in place of a linear programming relaxation to provide bounds in a branch and bound algorithm. It is also noted that after some heuristic adjustment of the Lagrange problem solution, this indicates the dual solution of the problem, a good approximation of the optimal solution of the primal problem can be obtained.

In our analysis, we transformed multiple objective functions into a single objective function which is called the scalarization technique for multiobjective optimization problems. The significance of the scalarization techniques is that once we transform the original into auxiliary problems, we can utilize the existing techniques used for single-objective optimization problems available in the literature. This paper proposes the Lagrangian relaxation problem for multiobjective optimization problems. Furthermore, we generate Lagrange multipliers that can be used in the algorithms to solve the proposed problems. We have used the Brute force technique to solve the problems, and Lagrangian parameters are generated by using the discretization technique with an equal step size.

In Section 2, we introduce Lagrangian relaxation technique, and in Section 3, multiobjective optimization problem and scalarization technique are discussed. Section 4 illustrates the use of Lagrangian relaxation method to solve scalarization problems.

2. Lagrangian Relaxation Technique

We formulate a linear programming problem:

$$\text{(P)} \quad Z = \min cx,$$

subject to,

$$Ax = b, \quad Px \le d, \quad x \ge 0 \quad \text{and integers,}$$

where x is $n \times 1$, b is $m \times 1$, d is $k \times 1$ and all other matrices have conformable dimensions.

We assume that the constraints of (P) have been partitioned into the two sets $Ax = b$ and $Px \le d$. Now, we introduce new objective function that includes the original objective cx and part of the constraints $Ax = b$, thus the Problem (P) transforms into auxiliary problem

$$\text{(AP)} \quad Z_D(u) = \min cx + u(Ax - b), \quad Px \le d \quad \text{and} \quad x \ge 0,$$

where $u=(u_1,\cdots,u_m)$ is a vector of Langrage multipliers. It is easier to solve the problem (AP) than solve it (P).

For convenience, we assume that the feasible set of (P) is

$$X^P=\{x \mid Ax=b, Px\le d, x\ge 0 \text{ and } x \text{ integers}\},$$

and the feasible set of Problem (AP).

$$X^{AP}=\{x \mid Px\le d, x\ge 0 \text{ and } x \text{ integers}\}, \text{ and } X^P\subseteq X^{AP}.$$

Then, $Z_D(u)$ is finite for all *u*. It is easy to extend the development when these assumptions are violated or when inequality constraints are included in the set to be dualized.

It is well known that $Z_D(u)\le Z$. This is easy to show by assuming an optimal solution x^* to (P) and obtain that

$$Z_D(u)\le cx^*+u(Ax^*-b)=Z.$$

The inequality in the above relation follows from the definition of $Z_D(u)$ and equality from $Z=cx^*$ and $Ax^*-b=0$.

If $Ax=b$ is replaced by $Ax\le b$ in (P), then we require $u\ge 0$ and the relation becomes.

$$Z_D(u)\le cx^*+u(Ax^*-b)\le Z.$$

as $Z=cx^*$, $u\ge 0$ and $Ax^*-b\le 0$. Similarly, for $Ax\ge b$ we require $u\le 0$ for $Z_D(u)\le Z$ to hold.

In general, it is not possible to guarantee to find *u* for which $Z_D(u)=Z$, but this frequently happens for particular problem cases. The fact that $Z_D(u)\le Z$ allows (AP) to be used in place of (P) to provide lower bounds in a branch and bound algorithm for (P). While this is the most obvious use of (AP), it has a number of other uses. It can be a medium for selecting branching variables and choosing the next branch to explore. Good feasible solutions to (P) can frequently be obtained by perturbing nearly feasible solutions to (AP). Finally, Lagrangian relaxation has been used recently in [5] as an analytic tool for establishing worst-case bounds on the performance of certain heuristics.

Determining Lagrangian Parameters *u*

One of the important steps is finding the suitable Lagrangian parameter *u* that provides the optimal solution of (AP).

The best choice of finding *u* is to solve the dual problem of (AP) which is stated as follows (see for details in [4]).

$$Z=\max w,$$

subject to,

$$w\le cx^t+u(Ax^t-b),\ t=1,\cdots,t,$$

$$u\ge 0,$$

where x^t is the solution of (AP).

Held, Wolfe and Crowder Approach (1974) [4]:

We are here recalling the main steps to calculate *u*. A detail of this setting can be seen in [4]. It is required to determine a sequence of values for u that are

$$u^{k+1} = \max\left\{0, u^k + t_k\left(Ax^k - b\right)\right\},$$

where t^k is the step-size and x^k are the optimal solution of (AP).

The step-size t^k can be determined as follows.

$$t^k = \frac{\lambda_k\left(Z^* - Z_D\left(u_k\right)\right)}{\left\|Ax^k - b\right\|^2}.$$

where Z^* is determined the value of known feasible solution. The sequence λ_k is concluded from the interval $[0,2]$ and suitable λ_k is needed if $Z_D\left(u_k\right)$ is failed to increase.

In this paper, we have not used the Held *et al.* approach. Instead, we generate a matrix for Lagrangian multipliers and solve the problem for each of the multipliers. In the end, we recommended the best possible multipliers that lead to getting optimum solution.

The following Table 1 depicts the chosen values of Lagrangian multipliers.

3. Multiobjective Optimization Problems

Many real-life problems that often deal with multiple objectives and intend to optimize all criteria simultaneously. We now formulate the multiobjective optimization problem as follows.

(MOP) minimize

$$f(\boldsymbol{x}) \equiv \left(f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_l(\boldsymbol{x})\right),$$

subject to the set

$$A = \left\{\boldsymbol{x} \in R^n \,\middle|\, \boldsymbol{g}(\boldsymbol{x}) \le 0\right\}.$$

Table 1. Lagrangian multipliers are generated with step size 0.1 within the interval $0 \le u \le 1$.

0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.01	0.11	0.21	0.31	**0.41**	0.51	0.61	0.71	0.81	0.91
0.02	0.12	0.22	0.32	0.42	0.52	0.62	0.72	0.82	0.92
0.03	0.13	0.23	0.33	0.43	0.53	0.63	0.73	0.83	0.93
0.04	0.14	0.24	0.34	0.44	0.54	0.64	0.74	0.84	0.94
0.05	0.15	0.25	0.35	0.45	0.55	0.65	0.75	0.85	0.95
0.06	0.16	0.26	0.36	0.46	0.56	0.66	0.76	0.86	0.96
0.07	0.17	0.27	0.37	0.47	0.57	0.67	0.77	0.87	0.97
0.08	0.18	0.28	0.38	0.48	0.58	0.68	0.78	0.88	0.98
0.09	0.19	0.29	0.39	0.49	0.59	0.69	0.79	0.89	0.99
									1

The solution of multiobjective optimization problem is called an efficient solution or weak efficient solution [6]. The relation between efficient and weak efficient is that every efficient point is a weak efficient point, but the reverse does not hold. There are a number of ways to solve (MOP), and one of the popular methods is the scalarization technique. When a multiobjective problem is solved by combining its multiple objectives into one single-objective scalar function, this approach is generally known as the scalarization method.

The *k*th objective weighted constraint problem introduced in [7] [8] is one of the well-known scalarization techniques. The method applies to the problems with the disconnected Pareto front and the problems with a disconnected feasible set under the mild assumptions that the objective functions are continuous and bounded from below with a known lower bound. For each fixed *k*, the *k*th objective is minimized while the other weighted objective functions are incorporated as constraints. The problem structure is as follows:

(Sc-MOP)

$$\min w_k f_k(x),$$

subject to

$$w_i f_i(x) \le w_k f_k(x),\ \ i = 1,\cdots,l;\, i \ne k,$$

$$w_i > 0,\ \ \sum\nolimits_{i=1}^{l} w_i = 1,$$

$$g_j(x) \le 0,\ j = 1,\cdots,n \ \ \text{and}\ \ w \in W := \left\{x \in E_l \mid w_i \ge 0, \sum\nolimits_{i=1}^{l} w_i = 1\right\}.$$

If $\bar{x}$ is a weak efficient point to the original problem (MOP) then $\bar{x}$ is the solution of the (Sc-MOP) for each fixed *k*. On the other hand, a point $\bar{x}$ solves all subproblems of (Sc-MOP) then it is an efficient solution to the original problem (MOP). This is a strong condition and it can be relaxed by setting $x_k, k = 1,\cdots,l$ solves the *k*-subproblems of (MOP), and if Proposition 3.3 in [5] holds, then $x_k, k = 1,\cdots,l$ are weak efficient solutions of the original problem (MOP).

4. Lagrangian Relaxation Technique for (MOP)

We now employ Lagrangian relaxation techniques to solve the subproblems of (Sc-MOP). We ease the constraints

$$w_i f_i(x) \le w_k f_k(x),\ \ i = 1,\cdots,l;\, i \ne k$$

We define an $(l-1)$ vector of nonnegative multipliers $p \in R^l, p \ge 0$ and added the nonnegative term

$$p_i\left(w_i f_i(x) - w_k f_k(x)\right),\ \ i = 1,\cdots,l;\, i \ne k,$$

to the objective function of (Sc-MOP). Therefore, the mathematical formulation of the revised *k*th subproblem is

(LR_{BMOP})

$$\min w_k f_k(x) + \sum\nolimits_{i=1}^{l-1} p_i\left[w_i f_i(x) - w_k f_k(x)\right],\ \ i \ne k,$$

subject to

$$w_i > 0,\ \sum_{i=1}^{l} w_i = 1,$$
$$g_j(x) \le 0,\ \ j = 1,\cdots,n,$$
$$p \ge 0.$$

This can be reformulated as

$$\min\left(1 - \sum_{i=1}^{l-1} p_i\right) w_k f_k(x) + \sum_{i=1}^{l-1} p_i w_i f_i(x),\ \ i \ne k,$$

subject to

$$w_i > 0,\ \sum_{i=1}^{l} w_i = 1,$$
$$g_j(x) \le 0,\ \ j = 1,\cdots,n,$$
$$p \ge 0.$$

To analyze the concepts, we now illustrate the following example.

Example 1. We take $n = l = 2$; Consider the problem

$$\min\{x_1, x_2\} \ \text{ and } \ X = \left\{(x_1, x_2) \middle| (x_1 - 1)^2 + (x_2 - 1)^2 - 1 \le 0\right\}.$$

Let's take $w_1 = 0.48$ and $w_2 = 0.52$.

We now employ Lagrangian relaxation approach in (Sc-MOP) to solve the Example 1.

(Subproblem-1)

$$\min w_1 f_1(x),$$

subject to

$$w_2 f_2(x) \le w_1 f_1(x),$$
$$w_i > 0,\ \sum_{i=1}^{2} w_i = 1.$$
$$x \in X.$$

We construct the auxiliary problem of Subproblem-1 using the Lagrangian relaxation technique given below.

(LR-1):

$$\min w_1 x_1 + p_2(w_2 x_2 - w_1 x_1),$$

subject to $x \in X$.

Similarly, (Subproblem-2)

$$\min w_2 f_2(x),$$

subject to

$$w_1 f_1(x) \le w_2 f_2(x),$$
$$w_i > 0,\ \sum_{i=1}^{2} w_i = 1,$$
$$x \in X.$$

Lagrangian relaxation technique of the above problem would be

(LR-2)

$$\min w_2 x_2 + p_1(w_1 x_1 - w_2 x_2),$$

subject to $x \in X$.

Solve (LR-1): The Lagrangian multiplier p_2 is provided into the algorithm. First, we form a matrix for p_2 under interval $0 \le p_2 \le 1$ that includes 101 parameter values with step-size is 0.1 (see Table 1). We solve (LR-1) for each multiplier, and the optimum solution is recorded. In the algorithm, the Brute Force technique is used to identify the minimum value of the problem. The implementation of the algorithm is done through MATLAB and it is also noted that the differentiable solvers "Ipopt" [9] and "SCIP" [10] are used to solve each of the subproblems. Associated solution-value for each of the multipliers is shown in Table 2.

Remark 1

The minimum value in Table 2 is found for $p_2 = 0$ (Lagrangian multiplier) and the respective solution is $x_1 = 0$ and $x_2 = 1$ which is shown in a square box (Yellow) in Figure 1. In Cell (1, 1) of Table 1 and Table 2 (Yellow in both tables) are the respective multiplier and the solution. However, $x_1 = 0$ and $x_2 = 1$ is not a feasible solution of (LR-1), this means, it violates $w_2 x_2 - w_1 x_1 \le 0$. Therefore, we need to integrate either Brute Force with constraint condition or Held *et al.* approach [4] in the algorithm to get the best *u* to obtain the optimal solution of (Subproblem-1). The optimal solution to this problem is $x_1 = 0.3072$ and $x_2 = 0.2789$ and the optimum value is 1.4163 (Green in both Table 1 and Table 2).

Figure 2 given below shows the optimal solution to (Subproblem-1) that also satisfies the constraint $w_2 x_2 - w_1 x_1 \le 0$.

Now we will solve the other subproblem (Subproblem 2) which is as follows

$$\min w_2 x_2 + p_1 \left(w_1 x_1 - w_2 x_2 \right),$$

subject to $x \in X$.

Similar Lagrangian multipliers (Table 1) are provided in the Algorithm to solve the problem (Subproblem 2). The solutions are obtained for associated

Table 2. Solution set of (LR-1) obtained for each multiplier stated in Table 1.

0	0.048882	0.090166	0.121552	0.140742	0.146164	0.137656	0.116551	0.085067	0.045545
0.005172	0.053388	0.09379	0.124054	**0.141919**	0.145928	0.13608	0.11383	0.08144	0.041232
0.010285	0.057816	0.097312	0.126431	0.142957	0.145553	0.134379	0.111008	0.077735	0.036861
0.015339	0.062164	0.100731	0.128682	0.143853	0.145039	0.132556	0.108086	0.073954	0.032434
0.020331	0.06643	0.104043	0.130805	0.144609	0.144387	0.130612	0.105068	0.0701	0.027952
0.025259	0.070611	0.107247	0.132796	0.145222	0.143598	0.128551	0.101956	0.066174	0.023417
0.030123	0.074705	0.110339	0.134656	0.145694	0.142674	0.126373	0.098752	0.062179	0.018831
0.034918	0.07871	0.113318	0.136382	0.146024	0.141616	0.124082	0.095459	0.058117	0.014194
0.039645	0.082624	0.116182	0.137972	0.146212	0.140426	0.12168	0.092079	0.053989	0.00951
0.0443	0.086443	0.118927	0.139426	0.146259	0.139105	0.119169	0.088614	0.049798	0.004778
									0

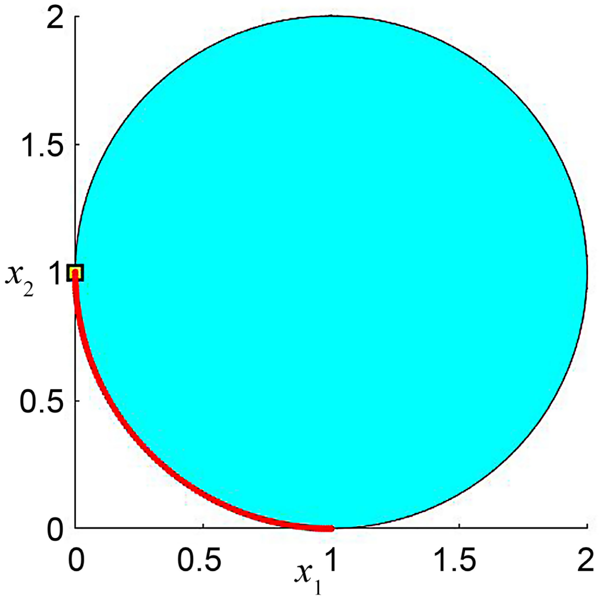

Figure 1. The square (Yellow) box depicts the optimum solution (but not feasible of (Subproblem-1)) and small circles are the solution of the respective other Lagrangian multipliers.

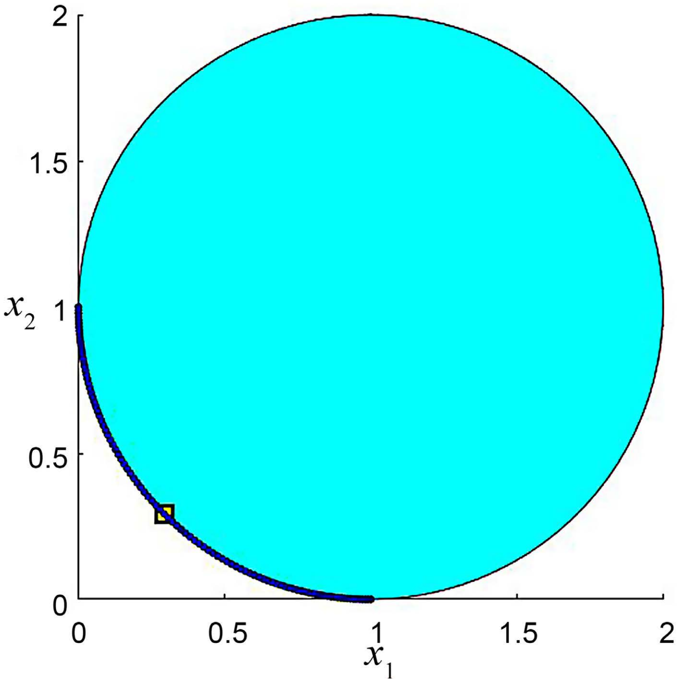

Figure 2. The square (Yellow) box depicts the optimum solution (which is also feasible of (Subproblem-1)) and small circles are the solution of the respective other Lagrangian multipliers.

multipliers. In the last step, the Brute Force method is applied to identify the optimum solution of (SubP2). **Figure 3** and **Table 3** are depicted as the solutions and optimum solution of the subproblem (Subproblem 2). The implementation required MATLAB programming software and app solvers.

Table 3. Solution list of (Subproblem 2) obtained for each multiplier listed in Table 1.

0	0.045545	0.085067	0.116551	0.137656	0.146164	0.140742	0.121552	0.090166	0.048882
0.004778	0.049798	0.088614	0.119169	0.139105	0.146259	0.139426	0.118927	0.086443	0.0443
0.00951	0.053989	0.092079	0.12168	0.140426	0.146212	0.137972	0.116182	0.082624	0.039645
0.014194	0.058117	0.095459	0.124082	0.141616	0.146024	0.136382	0.113318	0.07871	0.034918
0.018831	0.062179	0.098752	0.126373	0.142674	0.145694	0.134656	0.110339	0.074705	0.030123
0.023417	0.066174	0.101956	0.128551	0.143598	0.145222	0.132796	0.107247	0.070611	0.025259
0.027952	0.0701	0.105068	0.130612	0.144387	0.144609	0.130805	0.104043	0.06643	0.020331
0.032434	0.073954	0.108086	0.132556	0.145039	0.143853	0.128682	0.100731	0.062164	0.015339
0.036861	0.077735	0.111008	0.134379	0.145553	0.142957	0.126431	0.097312	0.057816	0.010285
0.041232	0.08144	0.11383	0.13608	0.145928	0.141919	0.124054	0.09379	0.053388	0.005172
									0

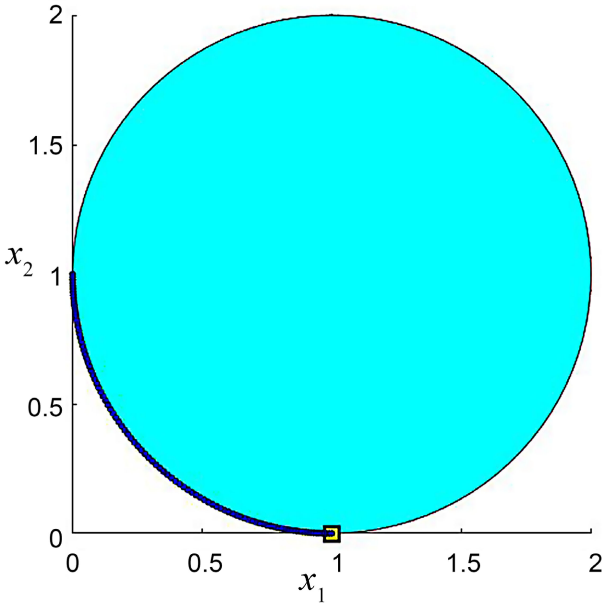

Figure 3. The square (Yellow) box depicts the optimum solution of (Subproblem 2) and small circles are the solution of the respective Lagrangian multipliers.

Remark 2

The optimum solution of (Subproblem 2) is found for $p_1 = 1$ (Lagrangian multiplier) and the solution is $x_1 = 1$ and $x_2 = 0$ which is shown in a square box (Yellow) in Figure 3. The minimum value is 0. In Cell (11, 10) of Table 1 and Table 3 (Green in both tables) are the respective multiplier and the solution. However, $x_1 = 1$ and $x_2 = 0$ is not a feasible solution of (Subproblem 2), which means, it violates constraints $w_2 x_2 - w_1 x_1 \leq 0$. Therefore, we need to fit in either the Brute Force with constraint condition or Held *et al.* [4] in the algorithm to get the best u to obtain the optimal solution of (Subproblem-2). The optimal solution (see, Figure 4) to this problem is $x_1 = 0.2929$ and $x_2 = 0.2929$ and the optimum value is 1.462 (Green in both Table 1 and Table 3).

Now we solve both subproblems for a set of weight vectors $w \in W$. This provides the solution of Example-1 and the solutions construct the Pareto front (see Figure 3) of the problem.

Remark 3

The implementation of Held *et al.* [4] algorithm is quite computationally expensive. Lambda which is in the interval $[0,2]$ needs to be chosen cautiously. This is required for each subproblem as well as each given weight. Figure 5 is given

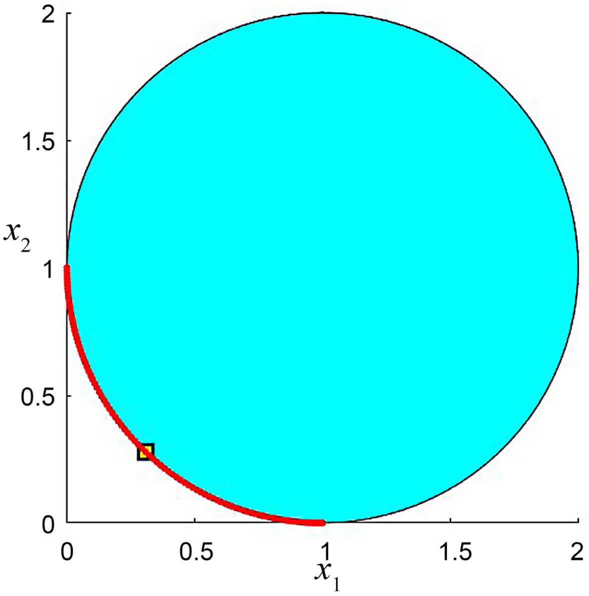

Figure 4. The square (Yellow) box depicts the optimum solution (which is also feasible of (Subproblem-2)), and small circles are the solution of the respective other Lagrangian multipliers.

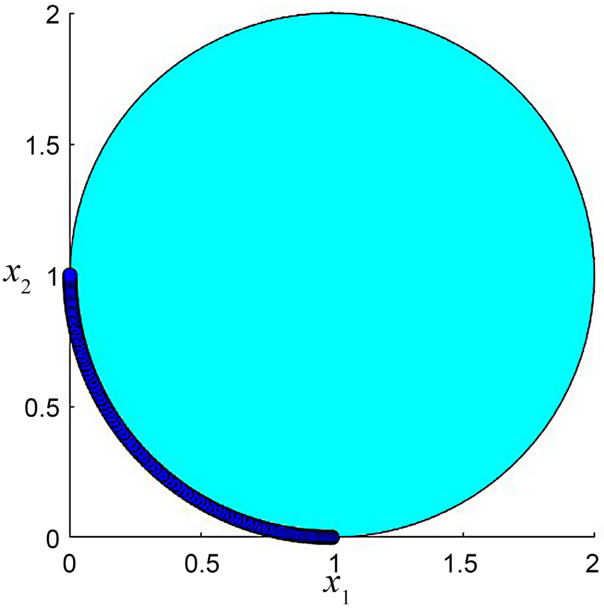

Figure 5. The circle (Blue) depicts the efficient solution of Example-1 which also approximates the Preto front.

below to demonstrate the Pareto front obtained by the Lagrangian relaxation approach (Subproblem 1) and (Subproblem 2).

5. Conclusion

The Lagrangian relaxation method to solve multiobjective optimization problems has been introduced in this paper. We began by selecting an appropriate and well-known scalarization method that transforms the original multiobjective into a scalar objective optimization problem. Afterwards, this scalar objective problem was solved by us using Lagrangian relaxation techniques. We supplied grids of multipliers in the algorithm followed by the application of Brute force techniques to sort optimum solutions. Ultimately, we analyzed the results and recommended efficient methods.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Fisher, M.L. (1981) The Lagrangian Relaxation Method for Solving Integer Programming Problems. *Management Science*, **27**, 1-18. https://doi.org/10.1287/mnsc.27.1.1

[2] Geoffrion, A.M. (1974) Lagrangian Relaxation for Integer Programming. *Mathematical Programming*, **2**, 82-114. https://doi.org/10.1007/BFb0120690

[3] Shapiro, J.F. (1979) Mathematical Programming: Structures and Algorithms. John Wiley, New York.

[4] Held, M., Wolfe, P. and Crowder, H.D. (1974) Validation of Subgradient Optimization. *Mathematical Programming*, **6**, 62-88. https://doi.org/10.1007/BF01580223

[5] Rizvi, M.M., Faruque, H.S.A. and Ray, G.C. (2021) Lagrangian Relaxation Method in Approximating the Pareto Front of Multiobjective Optimization Problems. *GANIT*: *Journal of Bangladesh Mathematical Society*, **40**, 126-133. https://doi.org/10.3329/ganit.v40i2.51315

[6] Xing, W. and Wu, B. (2012) Homotopy Continuous Method for Weak Efficient Solution of Multiobjective Optimization Problem with Feasible Set Unbounded Condition. *Applied Mathematics*, **3**, 765-771. https://doi.org/10.4236/am.2012.37114

[7] Burachik, R.S., Kaya, C.Y. and Rizvi, M.M. (2017) A New Scalarization Techniques and New Algorithms to Generate Pareto Fronts. *SIAM Journal on Optimization* (*SIOPT*), **27**, 1010-1034. https://doi.org/10.1137/16M1083967

[8] Burachik, R.S., Kaya, C.Y. and Rizvi, M.M. (2014) A New Scalarization Technique to Approximate Pareto Fronts of Problems with Disconnected Feasible Sets. *Journal of Optimization Theory and Applications*, **162**, 428-446. https://doi.org/10.1007/s10957-013-0346-0

[9] Wachter, A. and Biegler, L.T. (2006) On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-scale Nonlinear Programming. *Mathematical Programming*, **106**, 25-57. https://doi.org/10.1007/s10107-004-0559-y

[10] Gleixner, A., Eifler, L., Gally, T., Gamrath, G., Gemander, P., Gottwald, R.L., Hendel, G., Hojny, C., Koch, T., Miltenberger, M., Muller, B., Pfetsch, M.E., Puchert, E.,

Rehfeldt, D., Schlosser, F., Serrano, F., Shinano, Y., Viernickel, J.M., Vigerske, S., Weninger, D., Witt, J.T. and Witzig, J. (2017) The SCIP Optimization Suite 5.0. Zuse Institute Berlin, ZIB-Report 17-61, December.